On the Parameters of Convolutional Codes with Cyclic Structure

Heide Gluesing-Luerssen* and Barbara Langfeld†

February 1, 2008

Abstract

In this paper convolutional codes with cyclic structure will be investigated. These codes can be understood as left principal ideals in a suitable skew-polynomial ring. It has been shown in [3] that only certain combinations of the parameters (field size, length, dimension, and Forney indices) can occur for cyclic codes. We will investigate whether all these combinations can indeed be realized by a suitable cyclic code and, if so, how to construct such a code. A complete characterization and construction will be given for minimal cyclic codes. It is derived from a detailed investigation of the units in the skew-polynomial ring.

Keywords: Algebraic convolutional coding theory, cyclic convolutional codes, skew-polynomial rings, Forney indices.

MSC (2000): 94B10, 94B15, 16S36

1 Introduction

The two most important classes of codes used in practice are block codes and convolutional codes. While both classes play an equally important role in engineering practice, the theory of convolutional codes is much younger and not nearly as developed as the theory of block codes. The foundation of the mathematical theory of convolutional codes was laid only in the seventies of the last century by the articles of Forney, see e. g. [1]. It led to quite some mathematical investigation in that decade among which are basically two groups of papers.

The first group [11, 7, 8] deals with the construction of convolutional codes with large distance, mainly by using cyclic block codes and resorting to the weight-retaining property for bridging the gap between cosets of polynomials in the block code case and vector polynomials in the convolutional case. These ideas were resumed later again in [19], leading to the construction of MDS convolutional codes.

*Department of Mathematics, University of Kentucky, 715 Patterson Office Tower, Lexington, KY 40506, USA; heidegl@ms.uky.edu

†Kombinatorische Geometrie (M9), Zentrum Mathematik, Technische Universität München, Boltzmannstr. 3, 85747 Garching bei München, Germany; langfeld@ma.tum.de

The second group of papers [13, 14, 15] initiated a completely different approach. In the paper [14] it was investigated for the first time as to how cyclic structure has to be understood for a convolutional code itself. The first crucial fact being found was that cyclic structure in the classical sense (i. e. invariance under the cyclic shift) is not an appropriate concept for convolutional codes. Precisely, it was shown in [14] that each convolutional code, that is invariant under the cyclic shift, has complexity zero, hence is a block code. This insight has led Piret to a different, much more complex notion of cyclicity, which then was further generalized by Roos [15]. In the simplest form, this structure can be understood as a sort of graded shift in the coefficients of the polynomial codewords. The precise notion will be given in Section 2. At this point we only want to mention that cyclic codes of length n over the field $\mathbb{F}$ can be understood as certain left ideals in a skew-polynomial ring $A[z;\sigma]$, where $A = \mathbb{F}[x]/_{\langle x^n - 1\rangle}$, the variable z represents the delay operator, and σ determines the non-commutative structure. Both Piret and Roos gave several examples of convolutional codes, that are cyclic in this new sense. They also computed (or estimated) the distances which turned out to be very good.

Although these papers initiated an algebraic theory of cyclic convolutional codes, they did not come very far and the topic came to a halt. Only recently it has been resumed in [3]. Therein an algebraic theory of cyclic convolutional codes, fully in terms of ideals in the skew-polynomial ring, has been established. It leads to a nice, yet nontrivial, generalization of the algebraic theory of cyclic block codes. The translation from ideals into polynomial vectors is achieved by suitable circulant matrices. In particular, cyclic convolutional codes are principal left ideals (thus have a generator polynomial), they are also left annihilators of right ideals (thus have a parity check polynomial), the parameters can be computed in terms of these polynomials, and the dual of a cyclic code is cyclic again. Moreover, in [4] plenty of examples of cyclic convolutional codes are given, their distances are all optimal in the sense that they attain the Griesmer bound. All this indicates that the notion of cyclicity as introduced by Piret is the appropriate one for convolutional codes not only when it comes to the algebraic theory, but also for constructing good codes.

In this paper we will continue the algebraic theory as it was set up in [3]. It is a consequence of the results in [3] that only certain combinations of parameters (field size, length, dimension, and Forney indices) can occur for cyclic codes; see also Theorem 2.8(4) below. We seek to investigate whether all these combinations do really occur. The key role for this aim is played by so called minimal cyclic convolutional codes, these are cyclic codes that have no proper cyclic subcodes. They form the building blocks of all cyclic codes in the sense that each cyclic code is the direct sum of minimal codes and the Forney indices of the code are given by the union of the Forney indices of each component. Minimal codes have a very simple ideal theoretic description in terms of their generator polynomial, see Proposition 3.2. Moreover, for these codes all Forney indices are the same, hence these codes are compact in the sense of [12, Cor. 4.3]. This makes these codes also very important from a coding point of view since compact codes are in general good candidates for having a large distance. (for instance codes attaining the generalized Singleton bound are always compact, see [18]). We will show that under a certain necessary and sufficient condition any arbitrarily chosen Forney index can be realized by a suitable minimal cyclic code and we will show how to construct such a code. This result will then be further exploited for investigating non-minimal codes with prescribed Forney indices.

The outline of the paper is as follows. The end of the introduction is devoted to the basic notions of convolutional coding theory. Thereafter in Section 2 we will introduce cyclicity for convolutional codes along with the algebraic machinery and the main results from [3] as needed for our purposes. In Section 3 we turn to minimal cyclic convolutional codes. Their investigation amounts basically to a detailed study of the units in the skew polynomial ring $A[z;\sigma]$. This will lead us to the existence of minimal codes with prescribed Forney indices under certain necessary and sufficient conditions. Finally, in Section 4 we will turn to certain direct sums of minimal codes. These direct sums are specific in the sense that the generator polynomials of the minimal components are pairwise orthogonal, resulting in an easy handling of the direct sum. The existence result from Section 3 will be extended to these codes.

We will end the introduction with the basic notions of convolutional coding theory. Convolutional codes are certain submodules of $\mathbb{F}[z]^n$, where $\mathbb{F}$ is a finite field. Before presenting the definition we wish to recall that each submodule $\mathcal{S}$ of $\mathbb{F}[z]^n$ is free and therefore can be written as

$$\mathcal{S} = \operatorname{im} G := \big\{uG \,\big|\, u \in \mathbb{F}[z]^k\big\}$$

where k is the rank of $\mathcal{S}$ and $G \in \mathbb{F}[z]^{k\times n}$ is a matrix containing a basis of $\mathcal{S}$. Any such matrix G is called a *generator matrix* of the module $\mathcal{S}$. It is unique up to left multiplication by a unimodular matrix, that is, for any pair of matrices G, $G' \in \mathbb{F}[z]^{k\times n}$ having full row rank the identity $\operatorname{im} G = \operatorname{im} G'$ is equivalent to $G' = VG$ for some matrix $V \in Gl_k(\mathbb{F}[z])$. This makes the following notions well-defined.

Definition 1.1 Let $\mathbb{F}$ be any finite field and let $G \in \mathbb{F}[z]^{k\times n}$ be a matrix of rank k.

(a) The number $\delta := \delta(G) := \max\{\deg \gamma \mid \gamma \text{ is a } k\text{-minor of } G\}$ is called the *complexity* of the submodule $\operatorname{im} G$ or of the matrix G.

(b) The submodule $\mathcal{C} := \operatorname{im} G \subseteq \mathbb{F}[z]^n$ is called a *convolutional code over* $\mathbb{F}$ *with parameters* (n, k, δ) if it has complexity δ and the matrix G is right invertible, i. e. if there exists some matrix $\tilde{G} \in \mathbb{F}[z]^{n\times k}$ such that $G\tilde{G} = I_k$. In this case the parameter n is called the *length* of the code.

Since every right invertible matrix $G \in \mathbb{F}[z]^{k\times n}$ can be completed to a unimodular matrix (e.g. by using the Smith normal form) one has the following properties.

Remark 1.2 (a) The convolutional codes over $\mathbb{F}$ of length n are the direct summands of the module $\mathbb{F}[z]^n$.

(b) Each convolutional code $\mathcal{C} \subseteq \mathbb{F}[z]^n$ has a parity check matrix, that is, there exists a matrix $H \in \mathbb{F}[z]^{n\times(n-\operatorname{rk}\mathcal{C})}$ such that $\mathcal{C} = \ker H := \{v \in \mathbb{F}[z]^n \mid vH = 0\}$.

Part (b) can be considered as the main reason for restricting to direct summands rather than arbitrary submodules for convolutional codes. A parity check matrix is an important tool for data transmission, it is needed for checking whether or not the received data are erroneous.

The following property of convolutional codes will be needed later on.

Lemma 1.3 *Let $\mathcal{C}$, $\hat{\mathcal{C}} \subseteq \mathbb{F}[z]^n$ be two submodules having the same rank and satisfying $\hat{\mathcal{C}} \subseteq \mathcal{C}$. Furthermore, let $\hat{\mathcal{C}}$ be a convolutional code. Then $\hat{\mathcal{C}} = \mathcal{C}$.*

Proof: Let $\mathcal{C} = \operatorname{im} G$ and $\hat{\mathcal{C}} = \operatorname{im} \hat{G}$ where G, $\hat{G} \in \mathbb{F}[z]^{k\times n}$ and $\hat{G}$ is right invertible. The assumption $\hat{\mathcal{C}} \subseteq \mathcal{C}$ implies the existence of some matrix $U \in \mathbb{F}[z]^{k\times k}$ such that $\hat{G} = UG$. Using a right inverse of $\hat{G}$ shows that $U \in Gl_k(\mathbb{F}[z])$ and the assertion follows. □

The complexity is also known as the *overall constraint length* [6, p. 55], [1, p. 721] or the *degree* [12, Def. 3.5] of the code. It is an important parameter describing the size of the code and of the encoding process. In the coding literature a right invertible matrix is often also called *basic* [1, p. 730] or *delay-free and non-catastrophic*, see [12, p.1102]. Often in coding literature convolutional codes are defined as subspaces of the vector space $\mathbb{F}((z))^n$ of vector valued Laurent series over $\mathbb{F}$, see for instance [12] and [1]. However, as long as one restricts to right invertible generator matrices it makes no difference with respect to code properties and code constructions whether one works in the context of infinite message and codeword sequences (Laurent series) or finite ones (polynomials). Only for decoding it becomes important whether or not one may assume the sent codeword to be finite. The issue whether convolutional coding theory should be based on finite or infinite message sequences, has first been raised and discussed in detail in [17, 16].

It is well-known [1, Thm. 5] or [2, p. 495] that each submodule of $\mathbb{F}[z]^n$ has a minimal generator matrix in the sense of the next definition. In the same paper [2, Sec. 4] it has been shown how to derive such a matrix from a given generator matrix in a constructive way.

Definition 1.4 (1) For $v = \sum_{j=0}^{N} v_j z^j \in \mathbb{F}[z]^n$ where $v_j \in \mathbb{F}^n$ and $v_N \neq 0$ let $\deg v := N$ be the *degree* of v. Moreover, put $\deg 0 = -\infty$.

(2) Let $G \in \mathbb{F}[z]^{k\times n}$ be a matrix with rank k and complexity δ and let $\nu_1, \ldots, \nu_k$ be the degrees of the rows of G. We say that G is *minimal* if $\delta = \sum_{i=1}^{k} \nu_i$. In this case, the row degrees of G are uniquely determined by the submodule $\mathcal{S} := \operatorname{im} G$. They are called the *Forney indices* of $\mathcal{S}$.

The notion "minimal" stems from the (simple) fact that for an arbitrary generator matrix G one has $\delta \leq \sum_{i=1}^{k} \nu_i$. Thus, in a minimal generator matrix the rows degrees have been reduced to their minimal values.

From the above it follows that a convolutional code with parameters (n, k, δ) has a constant generator matrix if and only if $\delta = 0$. In that case the code can be regarded as an (n, k)-block code.

The most important concept for a code is its distance. It measures the error-correcting capability, hence the quality, of the code. The definition of the distance of a convolutional code is straightforward. For a constant vector $w = (w_1, \ldots, w_n) \in \mathbb{F}^n$ we define, just like in block code theory, its *(Hamming) weight* as $\operatorname{wt}(w) = \#\{i \mid w_i \neq 0\}$. For a polynomial vector $v = \sum_{j=0}^{N} v_j z^j \in \mathbb{F}[z]^n$, where $v_j \in \mathbb{F}^n$, the *weight* is defined as $\operatorname{wt}(v) = \sum_{j=0}^{N} \operatorname{wt}(v_j)$. Then the *(free) distance* of a code $\mathcal{C} \subseteq \mathbb{F}[z]^n$ with generator matrix $G \in \mathbb{F}[z]^{k\times n}$ is given as

$$\operatorname{dist}(\mathcal{C}) := \min\{\operatorname{wt}(v) \mid v \in \mathcal{C},\ v \neq 0\} = \min\left\{\operatorname{wt}(uG) \,\middle|\, u \in \mathbb{F}[z]^k,\ u \neq 0\right\}. \tag{1.1}$$

In coding theoretic terms, this notion is based on counting only the number of errors during data transmission, but not their magnitude; for more details about the distance of convolutional codes see for instance [6, Sec. 3.1]. Although we will not present any theoretical results concerning the distance of a cyclic convolutional code, we will show several examples of codes which do have optimal distance. In all these cases the distances have been computed with a computer algebra program and then compared to some suitable bound known from the literature. One of these bound is the *generalized Singleton bound* [18] stating that the distance d of a code with parameters (n,k,δ) over any field satisfies

$$d \leq S(n,k,\delta) := (n-k)\Big(\Big\lfloor\frac{\delta}{k}\Big\rfloor + 1\Big) + \delta + 1. \tag{1.2}$$

A code $\mathcal{C}$ with $\text{dist}(\mathcal{C}) = S(n,k,\delta)$ is called an MDS code. The *Griesmer bound* also takes the field size into account. It states that each code over a field with q elements and with parameters (n,k,δ) and largest Forney index m has distance d bounded by

$$d \leq \max\Big\{d' \in \{1,\ldots,S(n,k,\delta)\} \;\Big|\; \sum_{l=0}^{k(m+i)-\delta-1} \Big\lceil \frac{d'}{q^l} \Big\rceil \leq n(m+i) \text{ for all } i \in \hat{\mathbb{N}}\Big\}, \tag{1.3}$$

see [6, 3.22] for $q = 2$ and [4, Thm. 3.4] for general field size. Later we will present several codes where the distance attains this maximum value.

2 Cyclic Convolutional Codes

In this section we will introduce the notion of cyclicity for convolutional codes. After recalling from [14] that the classical notion of invariance under cyclic shift will always lead to complexity zero, we will introduce the skew-polynomial ring $A[z;\sigma]$, isomorphic to $\mathbb{F}[z]^n$ as left $\mathbb{F}[z]$-module, and call the codes corresponding to left ideals in $A[z;\sigma]$ cyclic. We will briefly discuss some features of $A[z;\sigma]$ and summarize the main results about cyclic codes, as obtained in [3], in Theorem 2.8. From this we will derive that cyclic codes always have a cyclic direct complement, thereby showing that the family of cyclic codes coincides with the family of those left ideals in $A[z;\sigma]$ that are direct summands.

Just like for cyclic block codes we assume from now on that

the length n and the field size $|\mathbb{F}|$ are coprime.

Recall that a block code $\mathcal{C} \subseteq \mathbb{F}^n$ is called cyclic if it is invariant under the cyclic shift, i. e.

$$(v_0,\ldots,v_{n-1}) \in \mathcal{C} \Longrightarrow (v_{n-1},v_0,\ldots,v_{n-2}) \in \mathcal{C} \tag{2.1}$$

for all $(v_0,\ldots,v_{n-1}) \in \mathbb{F}^n$. It is well-known that this is the case if and only if $\mathcal{C}$ is an ideal in the quotient ring

$$A := \mathbb{F}[x]/_{\langle x^n - 1\rangle} = \Big\{\sum_{i=0}^{n-1} f_i x^i \bmod (x^n-1) \;\Big|\; f_0,\ldots,f_{n-1} \in \mathbb{F}\Big\}, \tag{2.2}$$

canonically identified with $\mathbb{F}^n$ via

$$\mathfrak{p} : \mathbb{F}^n \longrightarrow A, \quad (v_0,\ldots,v_{n-1}) \longmapsto \sum_{i=0}^{n-1} v_i x^i \bmod (x^n-1).$$

Recall that the cyclic shift in $\mathbb{F}^n$ translates into multiplication by x in A, i. e.

$$\mathfrak{p}(v_{n-1}, v_0, \ldots, v_{n-2}) = x\mathfrak{p}(v_0, \ldots, v_{n-1}) \tag{2.3}$$

for all $(v_0, \ldots, v_{n-1}) \in \mathbb{F}^n$. It is well-known that each ideal $I \subseteq A$ is principal, hence there exists some $g \in A$ such that $I = \langle g \rangle$. One can even choose g as a monic divisor of $x^n - 1$, in which case it is usually called the *generator polynomial* of the code $\mathfrak{p}^{-1}(I) \subseteq \mathbb{F}^n$.

In order to extend the situation of cyclic block codes to the convolutional setting, we have to replace the vector space $\mathbb{F}^n$ by the free module $\mathbb{F}[z]^n$ and, consequently, the ring A by the polynomial ring $A[z]$ over A. Then we can extend the map $\mathfrak{p}$ above coefficient-wise to polynomials, thus

$$\mathfrak{p} : \mathbb{F}[z]^n \longrightarrow A[z], \quad \sum_{j=0}^{N} z^j v_j \longmapsto \sum_{j=0}^{N} z^j \mathfrak{p}(v_j), \tag{2.4}$$

where, of course, $v_j \in \mathbb{F}^n$ and thus $\mathfrak{p}(v_j) \in A$ for all j. This map is an isomorphism of $\mathbb{F}[z]$-modules. Its inverse will be denoted by

$$\mathfrak{v} := \mathfrak{p}^{-1}. \tag{2.5}$$

Again, by construction the cyclic shift in $\mathbb{F}[z]^n$ corresponds to multiplication by x in $A[z]$, that is, we have (2.3) for all $(v_0, \ldots, v_{n-1}) \in \mathbb{F}[z]^n$. At this point it is quite natural to call a convolutional code $\mathcal{C} \subseteq \mathbb{F}[z]^n$ cyclic if it is invariant under the cyclic shift, i. e. if (2.1) holds true for all $(v_0, \ldots, v_{n-1}) \in \mathbb{F}[z]^n$. This, however, does not result in any codes other than block codes due to the following result, see [14, Thm. 3.12] and [15, Thm. 6]. An elementary proof can be found at [3, Prop. 2.7].

Theorem 2.1 *Let $\mathcal{C} \subseteq \mathbb{F}[z]^n$ be a convolutional code with parameters (n, k, δ) such that* (2.1) *holds true for all $(v_0, \ldots, v_{n-1}) \in \mathbb{F}[z]^n$. Then $\delta = 0$, hence $\mathcal{C}$ is a block code.*

This result has led Piret [14] to suggesting a different notion of cyclicity for convolutional codes. We will present this notion in the slightly more general version introduced by Roos [15].

In order to do so notice that $\mathbb{F}$ can be regarded as a subfield of the ring A in a natural way. As a consequence, A is an $\mathbb{F}$-algebra. In the sequel the automorphism group $\mathrm{Aut}_{\mathbb{F}}(A)$ of the $\mathbb{F}$-algebra A will play an important role. It is clear that each automorphism $\sigma \in \mathrm{Aut}_{\mathbb{F}}(A)$ is uniquely determined by the single value $\sigma(x) \in A$. In particular, $\sigma(x) = x$ determines the identity map on A. But, of course, not every choice for $\sigma(x)$ determines an automorphism on A. Since x generates the $\mathbb{F}$-algebra A, the same has to be true for $\sigma(x)$ and, more precisely, we obtain for $a \in A$ that $\sigma(x) = a$ determines an automorphism on A if and only if $1, a, \ldots, a^{n-1}$ are linearly independent over $\mathbb{F}$ and $a^n = 1$. A better way to determine $\mathrm{Aut}_{\mathbb{F}}(A)$ will be described below in Remark 2.5.

The main idea of Piret was to impose a new ring structure on $A[z]$ and to call a code cyclic if it is a left ideal with respect to that ring structure. The new structure is non-commutative and based on an (arbitrarily chosen) automorphism on A. In detail, this looks as follows.

Definition 2.2 Let $\sigma \in \mathrm{Aut}_{\mathbb{F}}(A)$.

(1) On the set $A[z]$ we define addition as usual while multiplication is defined via the rule

$$\sum_{j=0}^{N} z^j a_j \cdot \sum_{l=0}^{M} z^l b_l = \sum_{t=0}^{N+M} z^t \sum_{j+l=t} \sigma^l(a_j) b_l \text{ for all } N,\, M \in \mathbb{N}_0 \text{ and } a_j,\, b_l \in A$$

along with classical multiplication for the coefficients in the quotient ring A. This turns $A[z]$ into a skew-polynomial ring, denoted by $A[z;\sigma]$. We also call $A[z;\sigma]$ a *Piret-algebra*.

(2) Consider the map $\mathfrak{p} : \mathbb{F}[z]^n \to A[z;\sigma]$ as in (2.4), where now the images $\mathfrak{p}(v) = \sum_{j=0}^{N} z^j \mathfrak{p}(v_j)$ are regarded as elements of $A[z;\sigma]$. A submodule $\mathcal{S} \subseteq \mathbb{F}[z]^n$ is said to be *σ-cyclic* if $\mathfrak{p}(\mathcal{S})$ is a left ideal in $A[z;\sigma]$. A convolutional code $\mathcal{C} \subseteq \mathbb{F}[z]^n$ is said to be *σ-cyclic* if $\mathcal{C}$ is a direct summand of $\mathbb{F}[z]^n$ and a σ-cyclic submodule.

A few comments are in order. First of all, notice that multiplication is determined by the rule

$$az = z\sigma(a) \text{ for all } a \in A \tag{2.6}$$

along with the rules of a non-commutative ring. Hence, unless σ is the identity, the indeterminate z does not commute with its coefficients. Consequently, it becomes important to distinguish between left and right coefficients of z. Of course, the coefficients can be moved to either side by applying the rule (2.6) since σ is invertible. In the sequel we will always use the representation via right coefficients since that is the one needed for the map $\mathfrak{p}$ in part (2) above. Since multiplication inside A remains the same as before A is a commutative subring of $A[z;\sigma]$. Moreover, since $\sigma|_{\mathbb{F}} = \mathrm{id}_{\mathbb{F}}$, the classical polynomial ring $\mathbb{F}[z]$ is a commutative subring of $A[z;\sigma]$, too. As a consequence, $A[z;\sigma]$ is a left and right $\mathbb{F}[z]$-module and the map $\mathfrak{p} : \mathbb{F}[z]^n \to A[z;\sigma]$ is an isomorphism of left $\mathbb{F}[z]$-modules (but not of right $\mathbb{F}[z]$-modules). In the special case where $\sigma = \mathrm{id}_A$, the ring $A[z;\sigma]$ is the classical commutative polynomial ring and we know from Theorem 2.1 that no σ-cyclic convolutional codes with nonzero complexity exist. Finally, it should be noted that cyclic block codes (in the classical sense of (2.1)) are σ-cyclic for all automorphisms σ.

It is also worth being noted that, due to the definition above, σ-cyclic convolutional codes are the left $A[z;\sigma]$-submodules of $A[z;\sigma]$ that are at the same time direct summands of the left $\mathbb{F}[z]$-module $A[z;\sigma]$. As it will turn out this implies that they are direct summands as $A[z;\sigma]$-modules. In other words, each σ-cyclic code has a direct complement that is σ-cyclic, too (see Corollary 2.9 below).

Example 2.3 Let us consider the case where $\mathbb{F} = \mathbb{F}_2$ and $n = 7$. Thus $A = \mathbb{F}_2[x]/_{\langle x^7 - 1\rangle}$. In this case $\mathrm{Aut}_{\mathbb{F}}(A)$ contains 18 automorphisms (see also [15, p. 680, Table II]), one of which is defined via $\sigma(x) = x^5$. We choose this automorphism for the following computations. Consider the polynomial

$$g := 1 + x^2 + x^3 + x^4 + z(x + x^2 + x^3 + x^5) + z^2(1 + x + x^4 + x^6) \in A[z;\sigma] \tag{2.7}$$

and denote by ${}^{\bullet}\langle g \rangle := \{fg \mid f \in A[z;\sigma]\}$ the left ideal generated by g in $A[z;\sigma]$. Moreover, put $\mathcal{C} := \mathfrak{v}({}^{\bullet}\langle g \rangle) \subseteq \mathbb{F}[z]^7$. We will show now that $\mathcal{C}$ is a direct summand of $\mathbb{F}[z]^7$, hence $\mathcal{C}$ is a σ-cyclic convolutional code. In order to do so we first notice that

$${}^{\bullet}\langle g \rangle = \mathrm{span}_{\mathbb{F}[z]}\{g,\, xg, \ldots, x^6 g\}$$

and therefore, using the isomorphism $\mathfrak{v}$ from (2.5),

$$\mathcal{C} = \left\{ uM \,\middle|\, u \in \mathbb{F}[z]^7 \right\} \text{ where } M = \begin{bmatrix} \mathfrak{v}(g) \\ \mathfrak{v}(xg) \\ \vdots \\ \mathfrak{v}(x^6 g) \end{bmatrix} \in \mathbb{F}[z]^{7\times 7}.$$

Thus we have to compute $x^i g$ for $i = 1, \ldots, 6$. Using the multiplication rule in (2.6) we obtain

$$\begin{aligned} xg &= x + x^3 + x^4 + x^5 + z(1 + x + x^3 + x^6) + z^2(x + x^3 + x^4 + x^5), \\ x^2 g &= x^2 + x^4 + x^5 + x^6 + z(x + x^4 + x^5 + x^6) + z^2(1 + x + x^2 + x^5), \\ x^3 g &= 1 + x^3 + x^5 + x^6 + z(x^2 + x^3 + x^4 + x^6) + z^2(1 + x^3 + x^5 + x^6) = g + x^2 g. \end{aligned}$$

Since $x^3 g$ is in the $\mathbb{F}$-span of the previous elements, we obtain ${}^{\bullet}\langle\, g\,\rangle = \text{span}_{\mathbb{F}[z]}\{g, xg, x^2 g\}$ and, since $\mathfrak{v}$ is an isomorphism, $\mathcal{C} = \left\{ uG \,\middle|\, u \in \mathbb{F}[z]^3 \right\}$, where

$$G = \begin{bmatrix} \mathfrak{v}(g) \\ \mathfrak{v}(xg) \\ \mathfrak{v}(x^2 g) \end{bmatrix} = \begin{bmatrix} 1+z^2 & z+z^2 & 1+z & 1+z & 1+z^2 & z & z^2 \\ z & 1+z+z^2 & 0 & 1+z+z^2 & 1+z^2 & 1+z^2 & z \\ z^2 & z+z^2 & 1+z^2 & 0 & 1+z & 1+z+z^2 & 1+z \end{bmatrix}.$$

One can easily check that the matrix G is right invertible and minimal (see Definition 1.4). Hence $\mathcal{C} \subseteq \mathbb{F}[z]^7$ is indeed a cyclic convolutional code. It is worth mentioning that $\text{dist}(\mathcal{C}) = 12$ (derived by a computer algebra program) and this is the optimum value for any convolutional code over $\mathbb{F}_2$ with parameters $(7, 3, 6)$ by virtue of the Griesmer bound (1.3).

In order to proceed with the theory of cyclic convolutional codes one needs some knowledge about the left ideals in the skew-polynomial ring $A[z;\sigma]$. In particular, we need to understand whether a given left ideal corresponds to a convolutional code rather than just to a submodule and, if so, if the parameters (dimension and complexity) can be recovered from the ideal. All this has been answered in the affirmative in [3]. In the sequel we will present the according results.

The main tool for describing the left ideals in $A[z;\sigma]$ is the fact that A is a semi-simple ring. Since we need the details of this fact we will first elaborate on this. By comprimeness of the length n and the field size $|\mathbb{F}|$, the polynomial $x^n - 1$ is square free, say

$$x^n - 1 = \pi_1 \cdot \ldots \cdot \pi_r, \tag{2.8}$$

where $\pi_1, \ldots, \pi_r \in \mathbb{F}[x]$ are irreducible, monic, and pairwise different. We will also assume that the polynomials are ordered according to

$$\deg_x \pi_1 = \ldots = \deg_x \pi_{r_1} < \ldots < \deg_x \pi_{r_1+\ldots+r_{s-1}+1} = \ldots = \deg \pi_{r_1+\ldots+r_s}, \tag{2.9}$$

where $r_1 + \ldots + r_s = r$. Using $r_0 := 0$ and $l_t := \sum_{\lambda=0}^{t-1} r_\lambda + 1$ for $t = 1, \ldots, s$, we have the partition $\{1, \ldots, r\} = R^{(1)} \cup \ldots \cup R^{(s)}$ where $R^{(t)} = \{l_t,\, l_t + 1, \ldots, l_t + r_t - 1\}$. It will also be convenient to use equivalence relation

$$k \equiv l :\Longleftrightarrow \deg_x \pi_k = \deg_x \pi_l. \tag{2.10}$$

Hence $k \equiv l$ if and only if k and l belong to the same index set $R^{(t)}$ for some t.

The Chinese Remainder Theorem provides us with an isomorphism of rings

$$\psi : A \longrightarrow K_1 \times \ldots \times K_r, \quad a \longmapsto [\![\rho_1(a), \ldots, \rho_r(a)]\!], \tag{2.11}$$

where $K_k = \mathbb{F}[x]/_{\langle \pi_k \rangle}$ and ρ_k denotes the canonical projection. Notice that $K_k \cong K_l$ if and only if $k \equiv l$. As indicated in (2.11), the elements in the direct product will be denoted by $[\![a_1, \ldots, a_r]\!]$. It is easy to see that the elements

$$\varepsilon^{(k)} := \psi^{-1}\big([\![(\delta_{kj})_{1 \leq j \leq r}]\!]\big) \text{ for } k = 1, \ldots, r$$

form the uniquely determined set of primitive idempotents in A. We call the subfield $K^{(k)} := \varepsilon^{(k)}A = \psi^{-1}(0 \times \ldots \times 0 \times K_k \times 0 \times \ldots \times 0)$ the k-th component of A. Obviously, $A = K^{(1)} \oplus \ldots \oplus K^{(r)}$, showing that A is a semisimple left-Artinian ring, see e. g. [5, Ch. IX, Sec. 3.1]. In particular, A has only finitely many ideals, each of which being isomorphic to a direct product of fields. Moreover,

$$a \in A \text{ is a unit in } A \Longleftrightarrow \varepsilon^{(l)}a \neq 0 \text{ for all } l = 1, \ldots, r. \tag{2.12}$$

Let us now study the effect of a given automorphism $\sigma \in \mathrm{Aut}_{\mathbb{F}}(A)$ on the components. It is straightforward to see that for each k we have $\sigma(K^{(k)}) = K^{(l)}$ for some l such that $l \equiv k$. In other words,

$$\sigma(\varepsilon^{(k)}) = \varepsilon^{(l)} \text{ for some } l \text{ such that } k \equiv l. \tag{2.13}$$

This gives rise to the following definition.

Definition 2.4 Let $\sigma \in \mathrm{Aut}_{\mathbb{F}}(A)$. Define the permutation $\Pi_\sigma \in S_r$ via $\Pi_\sigma(k) = l$ where l is such that $\sigma(\varepsilon^{(k)}) = \varepsilon^{(l)}$ for all $k = 1 \ldots, r$. We call Π_σ the permutation corresponding to σ. Furthermore, define the equivalence relation $\equiv_\sigma$ on the index set $\{1, \ldots, r\}$ via $k \equiv_\sigma l$ if there exists some $i \in \mathbb{N}_0$ such that $\sigma^i(\varepsilon^{(k)}) = \varepsilon^{(l)}$.

Of course, the permutation Π_σ simply reflects the permutation induced by σ on the set $\{\varepsilon^{(1)}, \ldots, \varepsilon^{(r)}\}$, that is, $\sigma(\varepsilon^{(k)}) = \varepsilon^{(\Pi_\sigma(k))}$. The equivalence relation $\equiv_\sigma$ can also be expressed as $k \equiv_\sigma l$ if and only if k and l belong to the same cycle of the permutation Π_σ. Since the permutation Π_σ satisfies $\Pi_\sigma(R^{(t)}) = R^{(t)}$ for all $t = 1, \ldots, r$, see (2.13), we obtain that each of its cycles is contained in one of the sets $R^{(t)}$. In other words

$$k \equiv_\sigma l \Longrightarrow k \equiv l \quad \text{ for all } k,\, l \in \{1, \ldots, r\}.$$

The consideration above provides us with an alternative way to compute the automorphisms on A.

Remark 2.5 It is straightforward to see that each permutation $\Pi \in S_r$ satisfying $\Pi(R^{(k)}) = R^{(k)}$ for all $k \in \{1, \ldots, r\}$ is the permutation Π_σ of an $\mathbb{F}$-automorphism σ on A. Hence σ is such that $\sigma(K^{(k)}) = K^{(\Pi(k))}$ for all $k = 1, \ldots, r$. Since there are, in general, many isomorphisms between $K^{(k)}$ and $K^{(\Pi(k))}$, the permutation Π does not completely determine the automorphism. Rather, we obtain all automorphisms σ on A satisfying $\Pi_\sigma = \Pi$ by fixing one isomorphism between $K^{(k)}$ and $K^{(\Pi(k))}$ and using the automorphism group $\mathrm{Aut}_{\mathbb{F}}(K^{(k)})$ for presenting the remaining ones. One can show that in this way one obtains all automorphisms on A, see [20]. With this consideration one can easily compute the cardinality of the automorphism group. Indeed, notice that $r_1!\cdots r_s!$ counts the number of all permutations Π satisfying $\Pi(R^{(t)})=R^{(t)}$ for all t. Since each k is in one of the sets $R^{(t)}=\{l_t,l_t+1,\ldots,l_t+r_t-1\}$ and $|\mathrm{Aut}_{\mathbb{F}}(K^{(l_t)})|=\deg_x\pi_{l_t}$ the above leads to $|\mathrm{Aut}_{\mathbb{F}}(A)|=(\deg_x\pi_{l_1})^{r_1}\cdots(\deg_x\pi_{l_s})^{r_s}r_1!\cdots r_s!$. For more details see [3, Sec. 3].

Having this description of the semi-simple ring A and its automorphisms available we will now fix some $\sigma\in\mathrm{Aut}_{\mathbb{F}}(A)$ and turn to the Piret-algebra $A[z;\sigma]$ over A. This ring is, of course, an A-module and as such semisimple (i. e. every A-submodule of $A[z;\sigma]$ is a direct summand), see [5, Ch. IX, Thm. 3.7]. However, for our investigation of σ-cyclic codes we need to understand the ring structure along with the left $\mathbb{F}[z]$-module structure. This has been worked out in detail in [3] and leads to the following.

Using $1=\varepsilon^{(1)}+\ldots+\varepsilon^{(r)}$ we can write each polynomial $f\in A[z;\sigma]$ in the form

$$f=f^{(1)}+\ldots+f^{(r)},\ \text{where } f^{(k)}:=\varepsilon^{(k)}f.$$

We call $f^{(k)}$ the *k-th component* of f. Furthermore, the set $T_f:=\big\{k\in\{1,\ldots,r\}\,\big|\,f^{(k)}\neq 0\big\}$ is called the *support* of f. From (2.6) it follows that $\varepsilon^{(k)}z^{\mu}=z^{\mu}\varepsilon^{(k')}$ for some k' such that $k\equiv_{\sigma}k'$. Therefore, each $f\in A[z;\sigma]$ can be written as an A-linear combination of the elements

$$z^{\mu}\varepsilon^{(k)},\ \mu\geq 0,\ k=1,\ldots,r. \tag{2.14}$$

We call these elements the *monomials* of $A[z;\sigma]$. In particular, the k-th component $f^{(k)}=\varepsilon^{(k)}f$ of f satisfies

$$f^{(k)}\in\operatorname{span}_A\{z^{\mu}\varepsilon^{(k')}\mid\mu\geq 0,\ k'\equiv_{\sigma}k\} \tag{2.15}$$

(where the span has to be understood with respect to right coefficients). Thus, the (right) coefficients of $f^{(k)}$ are not in $\varepsilon^{(k)}A$ but rather move around in the fields $K^{(k')}=\varepsilon^{(k')}A$, where $k'\equiv_{\sigma}k$. From this and the orthogonality of the idempotents it follows immediately the orthogonality of components corresponding to disjoint cycles, precisely

$$f,\,g\in A[z;\sigma],\ k\not\equiv_{\sigma}l\Longrightarrow f^{(k)}g^{(l)}=g^{(l)}f^{(k)}=0. \tag{2.16}$$

Example 2.6 Consider again Example 2.3 where $\mathbb{F}=\mathbb{F}_2$, $n=7$ and $\sigma(x)=x^5$. The polynomial x^7-1 decomposes into $x^7-1=\pi_1\pi_2\pi_3$ where

$$\pi_1=x+1,\ \pi_2=x^3+x+1,\ \pi_3=x^3+x^2+1.$$

Thus, in the notation of (2.8) and (2.9), $r=3$, $s=2$ and $R^{(1)}=\{1\}$ and $R^{(2)}=\{2,\,3\}$. Furthermore, one has the primitive idempotents

$$\varepsilon^{(1)}=1+x+x^2+x^3+x^4+x^5+x^6,\ \varepsilon^{(2)}=1+x+x^2+x^4,\ \varepsilon^{(3)}=1+x^3+x^5+x^6,$$

which can easily be checked by verifying that $(\varepsilon^{(k)}\bmod\pi_i)=\delta_{ik}$ for $i,\,k=1,\,2,\,3$. Moreover, $\sigma(\varepsilon^{(1)})=\varepsilon^{(1)}$, $\sigma(\varepsilon^{(2)})=\varepsilon^{(3)}$, $\sigma(\varepsilon^{(3)})=\varepsilon^{(2)}$. In other words, σ induces the permutation $\Pi_{\sigma}=(1)(2,3)$. It can straightforwardly be shown that the polynomial g given in (2.7) satisfies $g^{(1)}=0=g^{(2)}$ as well as

$$g=g^{(3)}=\varepsilon^{(3)}(1+x+x^2)+z\varepsilon^{(2)}x+z^2\varepsilon^{(3)}x.$$

Hence $\psi(g) = [\![0,0,1+x+x^2]\!] + z[\![0,x,0]\!] + z^2[\![0,0,x]\!]$. This can be verified directly and expresses the fact that the coefficient g_0 of z^0 in g satisfies $(g_0 \bmod \pi_1) = 0 = (g_0 \bmod \pi_2)$ and $(g_0 \bmod \pi_3) = 1+x+x^2$. According relations hold for the coefficients of z and z^2.

Having this description of the polynomials in the Piret-algebra $A[z;\sigma]$ at hand we are now in a position to investigate the left ideals. In [3] a Groebner-type theory has been established for $A[z;\sigma]$. It is based on the monomials given in (2.14) and leads to a reduction algorithm just like for commutative polynomials in several variables. This looks as follows.

Definition 2.7 (a) Given two monomials $z^\mu\varepsilon^{(k)}$ and $z^\nu\varepsilon^{(l)}$ we define

$$z^\mu\varepsilon^{(k)} < z^\nu\varepsilon^{(l)} \Longleftrightarrow \mu < \nu \text{ or } \mu = \nu \text{ and } k < l.$$

(b) For a polynomial $f = \sum_{\nu\geq 0} z^\nu f_\nu = \sum_{\nu\geq 0}\sum_{l=1}^{r} z^\nu\varepsilon^{(l)} f_\nu \in A[z;\sigma]$ define $LM(f)$ to be the largest monomial $z^\mu\varepsilon^{(k)}$ (with respect to $<$) which has a nonzero coefficient in f, that is, for which $\varepsilon^{(k)}f_\mu \neq 0$. We call $LM(f)$ the *leading monomial of* f. The summands $z^\nu\varepsilon^{(l)}f_\nu$ are called the *terms of* f.

(c) A polynomial $f \in A[z;\sigma]$ is called *(left) reduced* if for all k, $l = 1,\ldots,r$, where $k \neq l$, no nonzero term of $f^{(k)}$ is right divisible by $LM(f^{(l)})$.

(d) A polynomial $f \in A[z;\sigma]$ is called a *component* if $f = f^{(k)}$ for some $k = 1,\ldots,r$.

One easily verifies that $<$ is a well-ordering on the set of monomials with respect to multiplication as far as the result is nonzero. Notice that a component $f^{(k)}$ is always reduced.

In [3] a reduction procedure for polynomials has been established which, just like in the commutative case of several variables, leads in a constructive way to a type of Groebner bases for left ideals in $A[z;\sigma]$. We will need the following results on principal left ideals. They have been proven in [3, Thm. 4.5, Cor. 4.13(b), Prop. 7.10, Thm. 7.13].

Theorem 2.8 *Fix* $\sigma \in \mathrm{Aut}_{\mathbb{F}}(A)$. *Then*

(1) Each principal left ideal $\mathcal{I} \in A[z;\sigma]$ *has a reduced generator polynomial. Precisely, there exists a reduced polynomial* $g \in A[z;\sigma]$ *such that*

$$\mathcal{I} = {}^\bullet\langle\, g\,\rangle := \{fg \mid f \in A[z;\sigma]\}.$$

Moreover, the reduced generator is unique up to left multiplication by units in A.

(2) Let $\mathcal{C} \subseteq \mathbb{F}[z]^n$ *be a* σ*-cyclic convolutional code. Then the associated left ideal* $\mathfrak{p}(\mathcal{C})$ *is principal and thus has a reduced generator* $g \in A[z;\sigma]$. *Moreover, the support of* g *satisfies* $T_g = T_{g_0}$ *where* g_0 *denotes the constant term of* g.

(3) Let $g \in A[z;\sigma]$ *be a reduced polynomial. Then* $\mathfrak{v}({}^\bullet\langle\, g\,\rangle) \subseteq \mathbb{F}[z]^n$ *is a direct summand of* $\mathbb{F}[z]^n$ *(thus a* σ*-cyclic convolutional code) if and only if there exist* $a \in A$ *and a unit* $v \in A[z;\sigma]$ *such that* $g = av$.

(4) Let $g \in A[z;\sigma]$ *be a reduced polynomial with support* T_g. *For* $l \in T_g$ *let* $\deg_x \pi_l = \kappa_l$, *where* π_l *is as in* (2.8), *and put* $\kappa := \sum_{l\in T_g}\kappa_l$. *Then the matrix*

$$G := \left[\mathfrak{v}\big(x^i g^{(l)}\big)\right]_{l\in T_g,\, i=0,\ldots,\kappa_l-1} \in \mathbb{F}[z]^{\kappa\times n} \tag{2.17}$$

is a minimal generator matrix of the submodule $\mathcal{S} := \operatorname{im} G \subseteq \mathbb{F}[z]^n$. As a consequence, $\mathcal{S}$ is a submodule of rank κ and complexity $\delta = \sum_{l\in T_g} \kappa_l \deg_z g^{(l)}$. The Forney-indices are given by the numbers $\deg_z g^{(l)}$, $l \in T_g$, each one counted κ_l times.

We wish to comment on these results. First of all, it is worth mentioning that $A[z;\sigma]$ is not a left principal ideal ring. Part (2) above only states that left ideals associated to direct summands in $\mathbb{F}[z]^n$ are principal. Indeed, there exist left ideals that are not principal [3, Exa. 4.6(a)]. Secondly, as for part (3) above we wish to mention that each left inverse of some $v \in A[z;\sigma]$ is also a right inverse [3, p. 32]; this will slightly simplify the investigation of units. Due to zero divisors in the coefficient ring A, the skew-polynomial ring has plenty of units of higher z-degree, i. e., units, that are not in A. We will investigate this issue in more detail in the next section. Notice that a unit itself is never reduced unless it is in A, i. e., a constant. This follows for instance from (1) since a unit generates (as a left ideal) the full Piret-algebra $A[z;\sigma]$, which in turn has the reduced polynomial $1 \in A$ as a generator. Finally, we want to emphasize that according to (4) the parameters of σ-cyclic convolutional codes can occur only in certain combinations. In particular, the Forney indices appear, in general, with higher multiplicities depending on the degrees of the prime factors π_l. In the next section we will investigate this situation in more detail.

It is worth being stressed that part (2) and (3) above deal with direct summands of the left module $A[z;\sigma]$ over the ring $\mathbb{F}[z]$ and not over $A[z;\sigma]$. However, it can easily be deduced from the above that direct summands with respect to these different structures coincide. Indeed,

Corollary 2.9 *Let $\mathcal{I}$ be a left ideal in $A[z;\sigma]$. Then the following are equivalent*

(i) $\mathcal{I}$ is a direct summand of the left $\mathbb{F}[z]$-module $A[z;\sigma]$,

(ii) $\mathcal{I}$ is a direct summand of the left $A[z;\sigma]$-module $A[z;\sigma]$.

In particular, a σ-cyclic code has a direct summand that is σ-cyclic again. Furthermore, if $\mathcal{I}$ is a direct summand, then $\mathcal{I} = {}^{\bullet}\langle\, g\,\rangle$ where $g = \sum_{l\in T_g} u^{(l)}$ for some unit $u \in A[z;\sigma]$. In this case, a direct complement is given by ${}^{\bullet}\langle\, g'\,\rangle$ where $g' := \sum_{l\notin T_g} u^{(l)}$.

Before we give the proof we wish to add that, using the reduction procedure established in [3, Sec. 4], it is possible to test constructively whether or not a given reduced polynomial generates a direct summand. This also produces a direct summand in a constructive way.

Proof: The direction (ii) $\Rightarrow$ (i) is clear since each $A[z;\sigma]$-module is also an $\mathbb{F}[z]$-module.
(i) $\Rightarrow$ (ii): By Theorem 2.8(2) the ideal $\mathcal{I}$ is principal, say $\mathcal{I} = {}^{\bullet}\langle\, \hat{g}\,\rangle$, where $\hat{g}$ is a reduced polynomial. By part (3) of that theorem we have $\hat{g} = au$ for some $a \in A$ and a unit $u \in A[z;\sigma]$. Then $T_a = T_{\hat{g}}$. We can normalize the factor a in the following way. Since the ring A is a direct product of fields, there exists a unit $\hat{a} \in A$ such that $\hat{a}a = \sum_{l\in T_a} \varepsilon^{(l)}$. Hence $\mathcal{I} = {}^{\bullet}\langle\, g\,\rangle$ where $g := \hat{a}au = \sum_{l\in T_a} u^{(l)}$ and $T_g = T_a$. A direct complement is given by the left ideal ${}^{\bullet}\langle\, g'\,\rangle$ where $g' := \sum_{l\notin T_g} u^{(l)}$. In order to see this, notice first that $g + g' = u$ is a unit in $A[z;\sigma]$ and hence ${}^{\bullet}\langle\, g\,\rangle + {}^{\bullet}\langle\, g'\,\rangle = A[z;\sigma]$. Suppose now that $fg = f'g' \in {}^{\bullet}\langle\, g\,\rangle \cap {}^{\bullet}\langle\, g'\,\rangle$ for some f, $f' \in A[z;\sigma]$. Then $(f\sum_{l\in T_g} \varepsilon^{(l)} - f'\sum_{l\notin T_g} \varepsilon^{(l)})u = 0$ and, since u is a unit, $f\sum_{l\in T_g} \varepsilon^{(l)} = f'\sum_{l\notin T_g} \varepsilon^{(l)}$. But this implies $f\varepsilon^{(k)} = f\sum_{l\in T_g} \varepsilon^{(l)}\varepsilon^{(k)} = f'\sum_{l\notin T_g} \varepsilon^{(l)}\varepsilon^{(k)} = 0$

for all $k \in T_g$. Hence $fg = \sum_{k\in T_g} f\varepsilon^{(k)}g = 0$, showing that ${}^{\bullet}\langle\, g\,\rangle \cap {}^{\bullet}\langle\, g'\,\rangle = \{0\}$. All this also proves the additional assertion. $\Box$

The above shows that the set of σ-cyclic codes is the same as the set of direct summands of the ring $A[z;\sigma]$. In this context it is worth being recalled that in every ring R with 1, a left ideal $\mathcal{I}$, that is a direct summand (as left R-module), is left principal and even has an idempotent generator. We wish to emphasize that reduced generators, as guaranteed by Theorem 2.8(2), are in general not idempotent. But the corollary above shows how idempotent generators can easily be obtained from the reduced generator. Indeed, with the data as in the corollary we have that $g + g' = u$ is a unit in $A[z;\sigma]$. Thus $1 = u^{-1}g + u^{-1}g'$ and $u^{-1}gu^{-1}g' = u^{-1}g - u^{-1}gu^{-1}g \in {}^{\bullet}\langle\, g\,\rangle \cap {}^{\bullet}\langle\, g'\,\rangle = \{0\}$. From this it follows that both terms $u^{-1}g$ and $u^{-1}g'$ are idempotent generators of the respective left ideal. In general these idempotent generators have much higher degree than the reduced ones. At any rate, as Theorem 2.8(4) shows, the reduced generators are the more useful ones when it comes to the associated module in $\mathbb{F}[z]^n$.

Since the reduced generator of a principal left ideal is essentially unique, the following definition is well-posed.

Definition 2.10 Let $g \in A[z;\sigma]$ be a reduced polynomial. Then its support T_g is called the *support of the left ideal* ${}^{\bullet}\langle\, g\,\rangle$ and also the *support of the submodule* $\mathfrak{v}({}^{\bullet}\langle\, g\,\rangle)$.

The previous examples illustrate the results given so far.

Example 2.11 Let us return once more to Example 2.3 and its continuation in Example 2.6. In that case the polynomial $g = g^{(3)}$ is reduced since it is a component. It generates a left ideal corresponding to a code of rank $3 = \deg_x \pi_3$ and complexity $6 = \deg_x \pi_3 \deg_z g^{(3)}$ which has been given explicitly in Example 2.3. This is compliant with what has been stated in Theorem 2.8(4). A σ-cyclic direct complement of ${}^{\bullet}\langle\, g\,\rangle$ in $A[z;\sigma]$ is given by the left ideal generated by the polynomial

$$g' = x + x^3 + x^4 + z(1 + x^3 + x^5 + x^6).$$

One way to check this is by showing that $v = g + g' = 1 + x + x^2 + z(1 + x + x^2 + x^6) + z^2(1 + x + x^4 + x^6)$ is a unit in $A[z;\sigma]$. This is indeed the case, its inverse is given by $v^{-1} = 1 + x^2 + x^3 + x^6 + z(x + x^2) + z^2(1 + x^2 + x^5 + x^6)$. The components of v are given by

$$v^{(1)} = \varepsilon^{(1)},\ v^{(2)} = \varepsilon^{(2)}(1 + x + x^2) + z\varepsilon^{(3)},\ v^{(3)} = \varepsilon^{(3)}(x^2 + x + 1) + z\varepsilon^{(2)}x + z^2\varepsilon^{(3)}x$$

showing that $g = v^{(3)}$ while one easily verifies that $g' = v^{(1)} + v^{(2)}$. We do not discuss how one obtains such a direct complement, since that needs more detailed results from [3].

3 Minimal Cyclic Codes

As before, let $\mathbb{F}$ be a finite field such that n and $|\mathbb{F}|$ are coprime and let $\sigma \in \mathrm{Aut}_{\mathbb{F}}(A)$ be a fixed automorphism, where A is as in (2.2). In this section we will investigate the building blocks of σ-cyclic convolutional codes, the minimal cyclic codes. We will derive necessary and sufficient conditions for the automorphism σ to allow for σ-cyclic codes with arbitrarily prescribed Forney indices.

As we saw in Theorem 2.8(2) each σ-cyclic convolutional code $\mathcal{C} \subseteq \mathbb{F}[z]^n$ corresponds to a principal left ideal in $A[z;\sigma]$ which is generated by a reduced polynomial. We will call each such reduced generator a *generator polynomial* of the code $\mathcal{C}$. Furthermore, part (4) of that theorem shows that each σ-cyclic convolutional code can be presented as the direct sum of σ-cyclic codes with components as generator polynomials. Indeed, using the isomorphism $\mathfrak{p}$, Equation (2.17) translates into the direct sum

$${}^{\bullet}\!\langle\, g \,\rangle = \bigoplus_{l \in T_g} {}^{\bullet}\!\langle\, g^{(l)} \,\rangle$$

of left ideals in $A[z;\sigma]$. This leads to the following definition.

Definition 3.1 Let $\{0\} \neq \mathcal{C} \subseteq \mathbb{F}[z]^n$ be a σ-cyclic convolutional code with generator polynomial $g \in A[z;\sigma]$. Then $\mathcal{C}$ is called *minimal* if g is a component, i. e. if $g = \varepsilon^{(l)} g = g^{(l)}$ for some $l \in \{1, \ldots, r\}$.

The notion “minimal” (which is not related to minimal generator matrices) is justified by the following result.

Proposition 3.2 *Let $\mathcal{C} \subseteq \mathbb{F}[z]^n$ be a σ-cyclic convolutional code with generator polynomial $g \in A[z;\sigma]$. Then the following are equivalent.*

(i) $\mathcal{C}$ is minimal,

(ii) $\mathcal{C} \neq \{0\}$ and $\mathcal{C}$ contains no proper σ-cyclic subcodes. Precisely, if $\hat{\mathcal{C}}$ is a σ-cyclic convolutional code and $\{0\} \neq \hat{\mathcal{C}} \subseteq \mathcal{C}$, then $\hat{\mathcal{C}} = \mathcal{C}$.

(iii) There exists a unit $u \in A[z;\sigma]$ such that $g = u^{(l)}$ for some index l.

Proof: (i) ⇒ (ii): By assumption $0 \neq g = g^{(l)}$ for some index l. Let $\{0\} \neq \hat{\mathcal{C}}$ be a σ-cyclic convolutional code with generator polynomial $h \neq 0$ and let $\hat{\mathcal{C}} \subseteq \mathcal{C}$. Then ${}^{\bullet}\!\langle\, h \,\rangle \subseteq {}^{\bullet}\!\langle\, g \,\rangle$, thus $h = fg$ for some $f \in A[z;\sigma]$. This implies $h_0 = f_0 g_0$ for the constant terms of the polynomials. From Theorem 2.8(2) we know that $g_0 = g_0^{(l)} \neq 0$, hence $h_0 = f_0 \varepsilon^{(l)} g_0 = h_0^{(l)}$. Using again Theorem 2.8(2) we deduce that $T_h = T_{h_0} = \{l\}$. Thus $h = h^{(l)}$ and by Theorem 2.8(4) the codes $\hat{\mathcal{C}}$ and $\mathcal{C}$ have the same rank. From Lemma 1.3 we conclude that $\hat{\mathcal{C}} = \mathcal{C}$.
(ii) ⇒ (i): follows directly from Theorem 2.8(4) since each component of the generator polynomial gives a σ-cyclic subcode of $\mathcal{C}$.
The equivalence (i) ⇔ (iii) is clear with Corollary 2.9. □

In the sequel we will show which parameters (n, k, δ) a minimal σ-cyclic convolutional code can attain. From Theorem 2.8(4) and Proposition 3.2 we have the following situation.

Remark 3.3 (a) Any component $u^{(l)}$ of a unit $u \in A[z;\sigma]$ defines a minimal σ-cyclic code $\mathfrak{v}({}^{\bullet}\!\langle\, u^{(l)} \,\rangle)$ with parameters (n, k, dk) where $k = \deg_x \pi_l$ and $d = \deg_z u^{(l)}$.

(b) Any minimal σ-cyclic code in $\mathbb{F}[z]^n$ with support $\{l\}$ has parameters (n, k, dk) and Forney index d counted k times, where $k = \deg_x \pi_l$ and d is the degree of the l-th component of a unit in $A[z;\sigma]$.

Hence the question raised above amounts to investigating as to which degrees can occur for a given component of a unit in $A[z;\sigma]$. The case where the complexity is zero is, of course, known from block code theory. Indeed, for each $k \in \{\deg_x \pi_1, \ldots, \deg_x \pi_r\}$ there exists a cyclic block code with parameters (n, k), hence a σ-cyclic convolutional code with parameters $(n, k, 0)$ for any automorphism σ. This follows also immediately from Remark 3.3(a). The existence of σ-cyclic convolutional codes with nonzero complexity however, implies certain relations between the parameters and the automorphism. Indeed, we have

Lemma 3.4 *Let $\mathcal{C} \subseteq \mathbb{F}[z]^n$ be a minimal σ-cyclic code with generator polynomial $g = g^{(l)}$. Then $\mathcal{C}$ has complexity zero if and only if $g = g\varepsilon^{(l)}$. Furthermore, if $\mathcal{C}$ has nonzero complexity then $\sigma\big(\varepsilon^{(l)}\big) \neq \varepsilon^{(l)}$.*

Proof: If $\mathcal{C}$ has complexity zero, then, by Theorem 2.8(4), the polynomial g has degree zero, thus $g \in A$. But then $g = \varepsilon^{(l)}g = g\varepsilon^{(l)}$ follows from commutativity of A. Conversely, $g = \varepsilon^{(l)}g = g\varepsilon^{(l)}$ implies ${}^{\bullet}\!\langle\, g \,\rangle \subseteq {}^{\bullet}\!\langle\, \varepsilon^{(l)} \,\rangle$ and thus $\mathcal{C} \subseteq \mathfrak{v}({}^{\bullet}\!\langle\, \varepsilon^{(l)} \,\rangle)$. Both submodules are direct summands and by virtue of Theorem 2.8(4) they have the same rank. Thus, Lemma 1.3 implies $\mathcal{C} = \mathfrak{v}({}^{\bullet}\!\langle\, \varepsilon^{(l)} \,\rangle)$ and therefore has complexity zero. As for the last assertion, notice that the identity $\sigma(\varepsilon^{(l)}) = \varepsilon^{(l)}$ and the very definition of multiplication in the Piret-algebra implies that $\varepsilon^{(l)}$ is in the center of $A[z;\sigma]$. Hence $g = \varepsilon^{(l)}g = g\varepsilon^{(l)}$ and the code has complexity zero. □

As a consequence we have that for given parameters n and $|\mathbb{F}|$ a given automorphism $\sigma \in \mathrm{Aut}_{\mathbb{F}}(A)$ admits (minimal) σ-cyclic convolutional codes of positive complexity only if the permutation $\Pi_\sigma \in S_r$ is nontrivial. This in turn is possible only if at least one of the sets $R^{(t)}$ contains more than one element (see Definition 2.4) or in other words, if $x^n - 1$ has (at least) two prime factors of the same degree. Recall that one easily obtains the degrees of the prime factors of $x^n - 1$ by computing the cyclotomic cosets modulo n over $\mathbb{F}$, see [10, Ch. 7, § 5]. With different methods it has been shown in [15, Sec. VI] and in [3, Prop. 3.4] that the condition $\Pi_\sigma \neq \mathrm{id}$ is not only necessary but also sufficient for the existence of σ-cyclic codes with positive complexity. Our goal is to prove even more. We will show that for any $\sigma \in \mathrm{Aut}_{\mathbb{F}}(A)$ and any $l \in \{1, \ldots, r\}$ such that $\sigma(\varepsilon^{(l)}) \neq \varepsilon^{(l)}$ and for any $d \in \mathbb{N}$ there exists a minimal σ-cyclic code with parameters (n, k, kd) where $k = \deg_x \pi_l$. To this aim we need

Definition 3.5 Let $\sigma \in \mathrm{Aut}_{\mathbb{F}}(A)$ and $l \in \{1, \ldots, r\}$. We define the l-order of σ as $o_l(\sigma) := \min\{m \in \mathbb{N} \mid \sigma^m(\varepsilon^{(l)}) = \varepsilon^{(l)}\}$.

Using the permutation $\Pi_\sigma \in S_r$ associated with σ, the l-order can also be expressed as $o_l(\sigma) = \min\{m \in \mathbb{N} \mid \Pi_\sigma^m(l) = l\}$. In other words, the l-order of σ is the length of the cycle of Π_σ containing l; therefore

$$l \equiv_\sigma l' \Longrightarrow o_l(\sigma) = o_{l'}(\sigma). \tag{3.1}$$

With the following lemma we will establish the existence of certain simple units in $A[z;\sigma]$. They will suffice to show the existence of the desired minimal σ-cyclic codes. We will also obtain that each unit in $A[z;\sigma]$ can be expressed as a finite product of these simple units. In this sense we can construct, at least theoretically, all units of $A[z;\sigma]$ and thus, by Corollary 2.9, all σ-cyclic convolutional codes.

Lemma 3.6 *Let $\sigma \in \mathrm{Aut}_{\mathbb{F}}(A)$ with l-order $o_l := o_l(\sigma)$ where $l \in \{1, \ldots, r\}$.*

(a) Let $a \in A$ and $d \in \mathbb{N}$. Put $u_{d,a,l} := 1 + z^d a\varepsilon^{(l)} \in A[z;\sigma]$. Then

$$u_{d,a,l} \text{ is a unit in } A[z;\sigma] \Longleftrightarrow \begin{cases} a^{(l)} \neq -\varepsilon^{(l)}, & \text{if } d = 0, \\ a^{(l)} = 0 \text{ or } o_l \nmid d, & \text{if } d > 0. \end{cases}$$

If $u_{d,a,l}$ is a unit in $A[z;\sigma]$, then its inverse is given by $u_{d,-a,l}$. In this case we call $u_{d,a,l}$ an elementary unit.

(b) Any unit in $A[z;\sigma]$ can be written as a finite product of elementary units.

Proof: (a) If $d = 0$ then $u_{d,a,l} = 1 + a^{(l)}$ and the assertion follows from (2.12). Thus let $d > 0$. We may assume $a^{(l)} \neq 0$ for otherwise the assertion is trivial.
"$\Rightarrow$" Write $u := u_{d,a,l}$, for short. Since u is a unit, we know from Remark 3.3(a) that $\mathfrak{v}({}^{\bullet}\!\langle\, u^{(l)} \,\rangle)$ is a minimal σ-cyclic convolutional code and its complexity is given by $\deg_x \pi_l \deg_z u^{(l)}$. If $o_l \mid d$, then $\varepsilon^{(l)} z^d = z^d \varepsilon^{(l)}$ and thus $u^{(l)} = \varepsilon^{(l)} u = u\varepsilon^{(l)} = \varepsilon^{(l)} + z^d a^{(l)}$, hence $\deg_z u^{(l)} = d > 0$. But on the other side Lemma 3.4 implies that the complexity of $\mathfrak{v}({}^{\bullet}\!\langle\, u^{(l)} \,\rangle)$ is zero, a contradiction.
"$\Leftarrow$" Let $o_l \nmid d$. Then $\sigma^d(\varepsilon^{(l)}) \neq \varepsilon^{(l)}$ and thus $\sigma^d(\varepsilon^{(l)})\varepsilon^{(l)} = 0$. But then

$$u_{d,a,l} u_{d,-a,l} = (1 + z^d a\varepsilon^{(l)})(1 - z^d a\varepsilon^{(l)}) = 1,$$

completing the proof of (a).
(b) Let $u \in A[z;\sigma]$ be a unit. Then ${}^{\bullet}\!\langle\, u \,\rangle = A[z;\sigma]$ and thus $1 \in A$ is a reduced generator of ${}^{\bullet}\!\langle\, u \,\rangle$. In [3, Cor. 4.13(a) and its proof] it has been shown that the reduction of a single polynomial in $A[z;\sigma]$ can be described by left multiplication with suitable elementary units. In other words, there exist elementary units $u_1, \ldots, u_t \in A[z;\sigma]$ such that $1 = u_t \cdot \ldots \cdot u_1 u$ which proves the assertion. □

It should be noticed that from a coding theoretic point of view the elementary units are not desirable if d is big. Indeed, since the coefficients of $z, z^2, \ldots, z^{d-1}$ are zero, the same is true for the coefficients of any component $u^{(l)}$ and thus the code $\mathfrak{v}({}^{\bullet}\!\langle\, u^{(l)} \,\rangle)$ has small distance. This argument, of course, does not apply if $d = 1$ and we will proceed with that more specific case. These units are not only candidates for the construction of good codes but, as we will see next, will lead us to the existence of the desired minimal σ-cyclic codes. To this end, we will now construct units whose l-th component have a prescribed degree.

Corollary 3.7 *Let $\sigma \in \mathrm{Aut}_{\mathbb{F}}(A)$ and $l \in \{1, \ldots, r\}$ such that $\sigma(\varepsilon^{(l)}) \neq \varepsilon^{(l)}$. Then we have*

(1) For any $a \in A$ and any $i \in \mathbb{N}_0$ the element $u_a(i) := 1 + za\sigma^i(\varepsilon^{(l)})$ is an elementary unit in $A[z;\sigma]$. Its inverse is given by $u_{-a}(i)$.

(2) For any $d \in \mathbb{N}_0$ and any units $a_1, \ldots, a_d$ in A the polynomial $u := u_{a_1}(1) \cdot \ldots \cdot u_{a_d}(d)$ is a unit in $A[z;\sigma]$ and satisfies $\deg_z u^{(l)} = d = \deg_z u$.

Proof: (1) If $\deg_z u_a(i) = 0$ the assertion is trivial. Thus let us assume $\deg_z u_a(i) = 1$. Note that, with the notation of Lemma 3.6, $u_a(i) = u_{1,a,l'}$ where l' is such that $\sigma^i(\varepsilon^{(l)}) = \varepsilon^{(l')}$. From (3.1) we know that $o_l(\sigma) = o_{l'}(\sigma)$ and by assumption this number is bigger than 1. Thus $o_{l'}(\sigma) \nmid \deg_z u_a(i)$ and Lemma 3.6(a) implies the assertion.
(2) Without loss of generality let $d > 0$. Let $u := u_{a_1}(1) \cdot \ldots \cdot u_{a_d}(d)$ where $a_1, \ldots, a_d$ are

units in A. From part (a) we know that u is a unit in $A[z;\sigma]$ and has $\deg_z u \leq d$. In order to show that $\deg_z u = d$ we compute the z^d-term of u. It is given by

$$\begin{aligned}&\big(za_1\sigma(\varepsilon^{(l)})\big)\cdot\big(za_2\sigma^2(\varepsilon^{(l)})\big)\cdot\ldots\cdot\big(za_d\sigma^d(\varepsilon^{(l)})\big)\\&\quad= z^d\big(\sigma^{d-1}(a_1)\sigma^{d-2}(a_2)\cdot\ldots\cdot\sigma(a_{d-1})a_d\big)\big(\sigma^d(\varepsilon^{(l)})\cdot\ldots\cdot\sigma^d(\varepsilon^{(l)})\big)\\&\quad= z^d a\sigma^d(\varepsilon^{(l)}),\end{aligned}$$

where $a := \sigma^{d-1}(a_1)\sigma^{d-2}(a_2)\cdot\ldots\cdot\sigma(a_{d-1})a_d$. Since $a_1,\ldots,a_d$ are units in A the same is true for a. Thus $a\sigma^d(\varepsilon^{(l)}) \neq 0$ and we have $\deg_z u = d$. Finally, $\deg_z u^{(l)} = d$ since $\varepsilon^{(l)}z^d a\sigma^d(\varepsilon^{(l)}) = z^d a\sigma^d(\varepsilon^{(l)}) \neq 0$. □

We would like to mention that for the unit u thus constructed $\deg_z u^{(l')} < d$ whenever $l' \neq l$. This can easily be seen from the above.

The following theorem combines our findings about the existence of minimal σ-cyclic convolutional codes. The proof follows from Theorem 2.8(4), Lemma 3.4, and Corollary 3.7(2).

Theorem 3.8 *Let $\sigma \in \mathrm{Aut}_{\mathbb{F}}(A)$ and $l \in \{1,\ldots,r\}$. Put $k := \deg_x \pi_l$. Then the following are equivalent:*

(i) $\sigma(\varepsilon^{(l)}) \neq \varepsilon^{(l)}$.

(ii) For any $d \in \mathbb{N}_0$ one can construct a minimal σ-cyclic convolutional code with parameters (n,k,dk) and support $\{l\}$. The Forney indices of the code are all equal to d.

(iii) There exists a σ-cyclic convolutional code with nonzero complexity and support $\{l\}$.

Notice that the considerations so far do not lead to any insight about the quality of a minimal σ-cyclic convolutional code, that is, about the distance. The following examples, however, suggest that this construction is worth being investigated with respect to distance properties. The codes given below are all optimal with respect to their distance. As for the general situation, we wish to add that the codes constructed in Theorem 3.8(ii) are *compact*, which in this case (rank k dividing the complexity) means that the Forney indices are all the same [12, Cor. 4.3]. In general, compact codes are better candidates for good codes; for instance, codes attaining the generalized Singleton bound (1.2) are always compact [18, Proof of Thm. 2.2].

Example 3.9 We begin with the case $n = 3$ over $\mathbb{F} := \mathbb{F}_4 = \{0,1,\alpha,\alpha^2\}$ where $\alpha^2+\alpha+1 = 0$. Thus $A = \mathbb{F}[x]/_{\langle x^3-1\rangle}$ and we have the prime factor decomposition $x^3 - 1 = \pi_1\pi_2\pi_3$ where $\pi_1 = x+1$, $\pi_2 = x+\alpha$, and $\pi_3 = x+\alpha^2$. The corresponding primitive idempotents are

$$\varepsilon^{(1)} = x^2+x+1,\ \varepsilon^{(2)} = \alpha x^2+\alpha^2 x+1,\ \varepsilon^{(3)} = \alpha^2 x^2+\alpha x+1$$

as can readily be seen by verifying $(\varepsilon^{(i)} \bmod \pi_j) = \delta_{ij}$ for $i,\ j = 1,2,3$. We will use the automorphism $\sigma \in \mathrm{Aut}_{\mathbb{F}}(A)$ defined by $\sigma(x) = x^2$. One easily checks that $\sigma(\varepsilon^{(2)}) = \varepsilon^{(3)}$ and vice versa. Hence $\Pi_\sigma = (1)(2,3)$. We will construct minimal σ-cyclic codes with support $\{2\}$ by using the construction of units in Corollary 3.7 for $l = 2$. Choose the units

$$v_1 = u_1(1),\ v_2 = u_\alpha(2),\ v_3 = u_{\alpha^2}(3),\ v_4 = u_\alpha(4),\ v_5 = u_{\alpha^2}(5),\ v_6 = u_\alpha(6) \in A[z;\sigma]$$

and put $g^{(\delta)} := \varepsilon^{(2)}(v_1 \cdot \ldots \cdot v_\delta)$ for $\delta = 1, \ldots, 6$. From Corollary 3.7(2) we know that $\deg_z g^{(\delta)} = \delta$ and that $\mathcal{C}^{(\delta)} := \mathfrak{v}(\overset{\bullet}{\langle}\, g^{(\delta)} \,\rangle)$ is a σ-cyclic code with parameters $(3, 1, \delta)_4$. We used a computer algebra program and computed the distances of these codes which turn out to be very good in each case. Indeed, the respective distances are

$$\mathrm{dist}(\mathcal{C}^{(1)})=6,\ \mathrm{dist}(\mathcal{C}^{(2)})=9,\ \mathrm{dist}(\mathcal{C}^{(3)})=12,\ \mathrm{dist}(\mathcal{C}^{(4)})=14,\ \mathrm{dist}(\mathcal{C}^{(5)})=16,\ \mathrm{dist}(\mathcal{C}^{(6)})=18.$$

For $\delta = 1, \ldots, 5$ the distances attain the Griesmer bound (1.3), hence these codes are optimal (for $\delta = 1, 2, 3$ this is even the generalized Singleton bound (1.2)). For $\delta = 6$ the computed distance is just one less than the Griesmer bound, which in this case is 19. It should be added that, as to our knowledge, it is unknown whether there exists any code over $\mathbb{F}_4$ with parameters $(3, 1, 6)$ and distance 19. We think it is worth presenting these codes explicitly. Recall from Theorem 2.8(4) that $G^{(\delta)} := \mathfrak{v}(g^{(\delta)})$ is a generator matrix of $\mathcal{C}^{(\delta)}$. These matrices are given by

$$G^{(1)} = [z{+}1,\ \alpha z{+}\alpha^2,\ \alpha^2 z{+}\alpha], \qquad G^{(2)} = [\alpha z^2{+}z{+}1,\ z^2{+}\alpha z{+}\alpha^2,\ \alpha^2 z^2{+}\alpha^2 z{+}\alpha],$$

$$G^{(3)} = \begin{bmatrix} z^3{+}\alpha z^2{+}\alpha z{+}1 \\ \alpha z^3{+}z^2{+}\alpha^2 z{+}\alpha^2 \\ \alpha^2 z^3{+}\alpha^2 z^2{+}z{+}\alpha \end{bmatrix}^{\mathsf{T}}, \qquad G^{(4)} = \begin{bmatrix} \alpha z^4{+}z^3{+}z^2{+}\alpha z{+}1 \\ z^4{+}\alpha z^3{+}\alpha^2 z^2{+}\alpha^2 z{+}\alpha^2 \\ \alpha^2 z^4{+}\alpha^2 z^3{+}\alpha z^2{+}z{+}\alpha \end{bmatrix}^{\mathsf{T}},$$

$$G^{(5)} = \begin{bmatrix} z^5{+}\alpha z^4{+}\alpha z^3{+}z^2{+}z{+}1 \\ \alpha z^5{+}z^4{+}\alpha^2 z^3{+}\alpha^2 z^2{+}\alpha z{+}\alpha^2 \\ \alpha^2 z^5{+}\alpha^2 z^4{+}z^3{+}\alpha z^2{+}\alpha^2 z{+}\alpha \end{bmatrix}^{\mathsf{T}}, \quad G^{(6)} = \begin{bmatrix} \alpha z^6{+}z^5{+}z^4{+}\alpha z^3{+}\alpha^2 z^2{+}z{+}1 \\ z^6{+}\alpha z^5{+}\alpha^2 z^4{+}\alpha^2 z^3{+}\alpha z^2{+}\alpha z{+}\alpha^2 \\ \alpha^2 z^6{+}\alpha^2 z^5{+}\alpha z^4{+}z^3{+}z^2{+}\alpha^2 z{+}\alpha \end{bmatrix}^{\mathsf{T}}.$$

Example 3.10 Now we consider the case $n = 5$ over $\mathbb{F} = \mathbb{F}_4 = \{0, 1, \alpha, \alpha^2\}$. In this case $x^5 - 1 = \pi_1\pi_2\pi_3$ where $\pi_1 = x + 1$, $\pi_2 = x^2 + \alpha x + 1$, and $\pi_3 = x^2 + \alpha^2 x + 1$ and the corresponding primitive idempotents are

$$\varepsilon^{(1)} = x^4{+}x^3{+}x^2{+}x{+}1,\ \varepsilon^{(2)} = \alpha x^4{+}\alpha^2 x^3{+}\alpha^2 x^2{+}\alpha x,\ \varepsilon^{(3)} = \alpha^2 x^4{+}\alpha x^3{+}\alpha x^2{+}\alpha^2 x.$$

We choose the automorphism defined via $\sigma(x) = x^2$. Again it is easily seen that $\sigma(\varepsilon^{(2)}) = \varepsilon^{(3)}$ and vice versa. We will use Corollary 3.7 for $l = 2$ in order to construct minimal σ-cyclic codes with support $\{2\}$. We define

$$g^{(1)} := \varepsilon^{(2)} u_1(1),\ g^{(2)} := \varepsilon^{(2)} u_1(1) u_\alpha(2),\ g^{(3)} := \varepsilon^{(2)} u_1(1) u_\alpha(2) u_{\alpha^2}(3).$$

Then we know that $\deg_z g^{(m)} = m$ and that $\mathcal{C}^{(m)} := \mathfrak{v}(\overset{\bullet}{\langle}\, g^{(m)} \,\rangle)$ is a σ-cyclic code over $\mathbb{F}_4$ with parameters $(5, 2, 2m)$ for $m = 1, 2, 3$. Again we computed the distances and they turn out to be optimal in each case. In this case Theorem 2.8(4) implies that the generator matrix of $\mathcal{C}^{(m)}$ is made up by the two rows $\mathfrak{v}(g^{(m)})$ and $\mathfrak{v}(xg^{(m)})$. They are computed as

$$G^{(1)} = \begin{bmatrix} 0 & \alpha + \alpha^2 z & \alpha^2 + \alpha z & \alpha^2 + \alpha z & \alpha + \alpha^2 z \\ \alpha + \alpha z & \alpha^2 z & \alpha & \alpha^2 + \alpha^2 z & \alpha^2 + \alpha z \end{bmatrix},$$

$$G^{(2)} = \begin{bmatrix} 0 & \alpha + \alpha^2 z + \alpha^2 z^2 & \alpha^2 + \alpha z + z^2 & \alpha^2 + \alpha z + z^2 & \alpha + \alpha^2 z + \alpha^2 z^2 \\ \alpha + \alpha z + \alpha^2 z^2 & \alpha^2 z + z^2 & \alpha + z^2 & \alpha^2 + \alpha^2 z + \alpha^2 z^2 & \alpha^2 + \alpha z \end{bmatrix},$$

$$G^{(3)} = \begin{bmatrix} 0 & \alpha{+}z{+}\alpha^2 z^2{+}\alpha^2 z^3 & \alpha^2{+}\alpha^2 z{+}z^2{+}\alpha z^3 & \alpha^2{+}\alpha^2 z{+}z^2{+}\alpha z^3 & \alpha{+}z{+}\alpha^2 z^2{+}\alpha^2 z^3 \\ \alpha{+}\alpha^2 z{+}\alpha^2 z^2{+}\alpha z^3 & z{+}z^2{+}\alpha z^3 & \alpha{+}z^2{+}\alpha^2 z^3 & \alpha^2{+}z{+}\alpha^2 z^2 & \alpha^2{+}\alpha^2 z{+}\alpha^2 z^3 \end{bmatrix}.$$

The distances are $\text{dist}(\mathcal{C}^{(1)}) = 8$, $\text{dist}(\mathcal{C}^{(2)}) = 12$, and $\text{dist}(\mathcal{C}^{(3)}) = 16$, which is in each case the Griesmer bound (1.3) for codes over $\mathbb{F}_4$ with parameters $(5, 2, 2m)$.

Remark 3.11 In [4, Table II] some other sequences of codes over $\mathbb{F}_4$ with parameters $(3, 1, \delta)$ for $\delta = 1, \ldots, 5$ and $(5, 2, 2m), m = 1, 2, 3$ have been given. They have the same distances as the ones given in the previous two examples, hence are also optimal. It is worth being pointed out that those codes and the ones presented here are *not* strongly equivalent in the sense that we call two codes $\text{im}\, G$ and $\text{im}\, G'$ *strongly equivalent* if $G = G'PD$ where $P \in Gl_n(\mathbb{F})$ is a permutation matrix and $D \in Gl_n(\mathbb{F})$ is a nonsingular diagonal matrix. In other words, codes are strongly equivalent if they differ only by a permutation and a rescaling of the entries of the codewords. Strongly equivalent codes have, of course, the same parameters and the same distance. From a coding point of view they have the same properties and can therefore be identified. As a consequence, the two families of codes obtained in the examples above are significantly different from those constructed earlier.

4 Orthogonal Sums of Minimal Cyclic Codes

In this section we will extend the existence result from Theorem 3.8 to certain non minimal σ-cyclic codes. The main tool for this task is the orthogonality as stated in (2.16). It leads directly to the following lemma. This in turn will imply that the sum of minimal codes having pairwise orthogonal generator polynomials is direct. Again, let $\mathbb{F}$ be a finite field such that $|\mathbb{F}|$ and n are coprime and let $\sigma \in \text{Aut}_{\mathbb{F}}(A)$ be a fixed automorphism. We will make heavy use of the prime factor decomposition (2.8) and the notations introduced in Definition 2.4.

Lemma 4.1 *Let $l_1, \ldots, l_t \in \{1, \ldots, r\}$ be such that $l_i \not\equiv_\sigma l_j$ for $i \neq j$. Furthermore, put $I := \{1, \ldots, r\} \backslash \{l \mid l \equiv_\sigma l_i$ for some $i = 1, \ldots, t\}$.*

(1) Let $u \in A[z; \sigma]$ be a unit with inverse $u^{-1} = \bar{u}$. Then

$$\sum_{j \equiv_\sigma l_i} u^{(j)} \sum_{j \equiv_\sigma l_i} \bar{u}^{(j)} = \sum_{j \equiv_\sigma l_i} \varepsilon^{(j)} \text{ for } i = 1, \ldots, t \text{ and } \sum_{j \in I} u^{(j)} \sum_{j \in I} \bar{u}^{(j)} = \sum_{j \in I} \varepsilon^{(j)}.$$

(2) For $i = 1, \ldots, t$ let $u_i \in A[z; \sigma]$ be a unit with inverse $u_i^{-1} = \bar{u}_i$ and let $u \in A[z; \sigma]$ be a unit with inverse $u^{-1} = \bar{u}$. Then the element $w := \sum_{i=1}^t \sum_{j \equiv_\sigma l_i} u_i^{(j)} + \sum_{j \in I} u^{(j)}$ is a unit with inverse $w^{-1} = \sum_{i=1}^t \sum_{j \equiv_\sigma l_i} \bar{u}_i^{(j)} + \sum_{j \in I} \bar{u}^{(j)}$.

(3) Each polynomial $g \in A[z; \sigma]$ with support $T_g = \{l_1, \ldots, l_t\}$ is reduced.

Proof: (1) The implication in (2.16) yields

$$u\bar{u} = \sum_{i=1}^t \Big(\sum_{j \equiv_\sigma l_i} u^{(j)} \sum_{j \equiv_\sigma l_i} \bar{u}^{(j)} \Big) + \sum_{j \in I} u^{(j)} \sum_{j \in I} \bar{u}^{(j)} = 1 = \sum_{j=1}^r \varepsilon^{(j)}.$$

From this the assertion follows immediately since the coefficients of each of the first t summands are contained in $\sum_{j \equiv_\sigma l_i} \varepsilon^{(j)} A$ while those of the second sum are in $\sum_{j \in I} \varepsilon^{(j)} A$ and all these sets are disjoint.
(2) follows from (1) along the same line of arguments.

(3) Write $g = \sum_{i=1}^{t} g^{(l_i)}$. By (2.15) the coefficients of z in $g^{(l_i)}$ are contained in $\sum_{l \equiv_\sigma l_i} \varepsilon^{(l)} A$ for all $i = 1, \ldots, t$. But then no term of some component $g^{(l_i)}$ can be right divisible by the leading monomial of any other component. □

All this leads to the existence of units with prescribed degrees for pairwise orthogonal components.

Theorem 4.2 *Let $l_1, \ldots, l_t \in \{1, \ldots, r\}$ be such that $l_i \not\equiv_\sigma l_j$ for $i \neq j$. Furthermore assume $o_{l_i}(\sigma) > 1$, that is, $\sigma(\varepsilon^{(l_i)}) \neq \varepsilon^{(l_i)}$, for all $i = 1, \ldots, t$. Then for all $d_1, \ldots, d_t \in \mathbb{N}_0$ there exists a unit $w \in A[z;\sigma]$ such that $g := \sum_{i=1}^{t} w^{(l_i)}$ is reduced and $\deg_z w^{(l_i)} = d_i$ for $i = 1, \ldots, t$.*

Proof: From Corollary 3.7(2) we know that for each $i = 1, \ldots, t$ there exists a unit u_i such that $\deg_z u_i^{(l_i)} = d_i$. Put $w := \sum_{i=1}^{t} \sum_{j \equiv_\sigma l_i} u_i^{(j)} + \sum_{i \in I} u_1^{(i)}$ where, again, $I = \{1, \ldots, r\} \backslash \{l \mid l \equiv_\sigma l_i \text{ for some } i = 1, \ldots, t\}$. Then Lemma 4.1(2) and (3) yield the desired results. □

Using Theorem 2.8(4) we obtain immediately the existence of orthogonal sums of minimal cyclic codes with prescribed Forney indices.

Corollary 4.3 *Let $l_1, \ldots, l_t \in \{1, \ldots, r\}$ be such that $l_i \not\equiv_\sigma l_j$ for $i \neq j$ and such that $o_{l_i}(\sigma) > 1$ for all $i = 1, \ldots, t$. Put $k_i := \deg_x \pi_{l_i}$. Then for all $d_1, \ldots, d_t \in \mathbb{N}_0$ there exists a σ-cyclic code $\mathcal{C} \subseteq \mathbb{F}[z]^n$ with parameters (n, k, δ) where $k = \sum_{i=1}^{t} k_i$ and $\delta = \sum_{i=1}^{t} k_i d_i$. The support is given by $\{l_1, \ldots, l_t\}$.*

Note that, according to Theorem 2.8(4), any σ-cyclic code with support $\{l_1, \ldots, l_t\}$ has to have parameters of the type above.

The arguments above may be used to construct non-minimal codes with given parameters and support consisting of indices with pairwise disjoint cycles directly out of minimal codes. We formulate the result in terms of direct summands in $\mathbb{F}[z]^n$.

Theorem 4.4 *For $i = 1, \ldots, t$ let $\mathcal{C}_i \subseteq \mathbb{F}[z]^n$ be a minimal σ-cyclic code with support $\{l_i\}$ and complexity δ_i and assume $l_i \not\equiv_\sigma l_j$ for $i \neq j$. Then $\mathcal{C} := \sum_{i=1}^{t} \mathcal{C}_i \subseteq \mathbb{F}[z]^n$ is a σ-cyclic code, too. Its rank is given by $\operatorname{rank}\mathcal{C} = \sum_{i=1}^{t} \operatorname{rank}\mathcal{C}_i = \sum_{i=1}^{t} \deg_x \pi_{l_i}$, and its complexity is $\delta(\mathcal{C}) = \delta_1 + \ldots + \delta_t$. Furthermore, $\mathcal{C} = \oplus_{i=1}^{t} \mathcal{C}_i$ and its Forney indices are given by the union of the Forney indices of the codes $\mathcal{C}_1, \ldots, \mathcal{C}_t$.*

Proof: For all $i = 1, \ldots, t$ let $\mathcal{C}_i = \mathfrak{v}(\overset{\bullet}{\langle}\, g_i \,\rangle)$ where $g_i = u_i^{(l_i)}$ for some unit $u_i \in A[z;\sigma]$. Put $g := g_1 + \ldots + g_t$ and $\mathcal{C} := \mathfrak{v}(\overset{\bullet}{\langle}\, g \,\rangle)$. Then, by Lemma 4.1(3), the polynomial g is reduced, and by part (2) of that lemma $g = \sum_{i=1}^{t} w^{(l_i)}$ for some suitable unit $w \in A[z;\sigma]$. Hence, by Theorem 2.8(3), the submodule $\mathcal{C}$ is a direct summand, and by part (4) of that theorem it it is the direct sum of $\mathcal{C}_1, \ldots, \mathcal{C}_t$ and has the desired rank, complexity, and Forney indices. □

We wish to illustrate the above by an example indicating that this construction does indeed lead to good codes.

Example 4.5 Let $n = 7$ and $\mathbb{F} = \mathbb{F}_8 = \{0, 1, \alpha, \alpha^2, \ldots, \alpha^6\}$ where $\alpha^3 + \alpha + 1 = 0$. Then $x^7 - 1 = \prod_{i=0}^{6} \pi_i$, where $\pi_i = x - \alpha^i$. Notice that, since all fields $K^{(i)} = \mathbb{F}[x]/_{\langle \pi_i \rangle}$ are isomorphic to $\mathbb{F}_8$, the automorphisms on $A = \mathbb{F}[x]/_{\langle x^7 - 1 \rangle}$ are fully determined by the permutation Π_σ. We choose the automorphism σ corresponding to the permutation $\Pi_\sigma = (1, 2)(3, 4, 5)(6)(7)$.

Moreover, we take the polynomials $g_1 = \varepsilon^{(1)} + z\varepsilon^{(2)} + z^2\varepsilon^{(1)}\alpha$ and $g_2 = \varepsilon^{(3)} + z\varepsilon^{(4)}\alpha + z^2\varepsilon^{(5)}\alpha^2$. Then $g_1 = \varepsilon^{(1)}g_1$ and $g_2 = \varepsilon^{(3)}g_2$. Since both polynomials, being components, are reduced, Theorem 2.8(4) tells us that $\overset{\bullet}{\langle}\, g_1 \rangle$ and $\overset{\bullet}{\langle}\, g_2 \rangle$ are submodules of rank 1 and complexity 2 each. It can be checked via some tedious but straightforward calculation that the associated matrices $\mathfrak{v}(g_i)$ are right invertible, thus both ideals are direct summands of $A[z;\sigma]$. Hence they are σ-cyclic codes over $\mathbb{F}_8$ with parameters $(7,1,2)$ each. Since $1 \not\equiv_\sigma 3$, the polynomial $g = g_1 + g_2$ is reduced (see Lemma 4.1(3)) and $\overset{\bullet}{\langle}\, g \rangle$ is a direct summand according to Theorem 4.4. A minimal generator matrix of the code $\mathfrak{v}(\overset{\bullet}{\langle}\, g \rangle) \subseteq \mathbb{F}_8[z]^7$ is given by

$$\begin{bmatrix} 1+z+\alpha z^2 & 1+\alpha^6 z+\alpha z^2 & 1+\alpha^5 z+\alpha z^2 & 1+\alpha^4 z+\alpha z^2 & 1+\alpha^3 z+\alpha z^2 & 1+\alpha^2 z+\alpha z^2 & 1+\alpha z+\alpha z^2 \\ 1+\alpha z+\alpha^2 z^2 & \alpha^5+\alpha^5 z+\alpha^5 z^2 & \alpha^3+\alpha^2 z+\alpha z^2 & \alpha+\alpha^6 z+\alpha^4 z^2 & \alpha^6+\alpha^3 z+z^2 & \alpha^4+z+\alpha^3 z^2 & \alpha^2+\alpha^4 z+\alpha^6 z^2 \end{bmatrix}.$$

The first and second row generate the codes $\mathfrak{v}(\overset{\bullet}{\langle}\, g_1 \rangle)$ and $\mathfrak{v}(\overset{\bullet}{\langle}\, g_2 \rangle)$, respectively. Again, all codes involved are optimal with respect to their distance. Both the codes $\mathfrak{v}(\overset{\bullet}{\langle}\, g_i \rangle)$, $i = 1,2$, have distance 21, which is the generalized Singleton bound (1.2). Hence these codes are MDS codes in the sense of [18]. The code $\mathfrak{v}(\overset{\bullet}{\langle}\, g \rangle)$ has distance 18, which is the optimum value for codes over $\mathbb{F}_8$ with parameters $(7,2,4)$ due to the Griesmer bound (1.3).

Finally we wish to comment on the existence of cyclic codes with arbitrary support. We will briefly sketch that the existence result of Corollary 4.3 is not true without the assumption $l_i \not\equiv_\sigma l_j$ for $i \neq j$. More precisely, in general it is not possible to arbitrarily prescribe the degrees of the components of a reduced polynomial. In order to see this, we consider a reduced polynomial g with support T_g containing at least two indices belonging to the same cycle of Π_σ. Without restriction assume $S = \{1,\ldots,c\} \subseteq T_g$ and $\sigma(\varepsilon^{(i)}) = \varepsilon^{(i+1)}$ for all $i = 1,\ldots,o-1$ where $o := o_1(\sigma) \geq c$. Let $\deg_z g^{(l)} = d_l$. Then for $l = 1,\ldots,c$ the highest coefficient of $g^{(l)}$ is in $\sigma^{d_l}(\varepsilon^{(l)})A = \varepsilon^{((l+d_l-1 \bmod o)+1)}A$ (the exponents arise from the fact that we have to compute modulo o with remainders in $\{1,\ldots,o\}$ instead of $\{0,\ldots,o-1\}$). Hence the reducedness of g implies that the numbers

$$(1+d_1),\ldots,(c+d_c) \text{ are pairwise different modulo } o.$$

But for $c > 1$ this puts a restriction on the degrees d_l of the components $g^{(l)}$ (even without using the fact that g is the generator polynomial of a code, i. e., of a direct summand). In case $c = o$, a second restriction arises if g generates a σ-cyclic code. In that case not all d_l can be the same for otherwise one can easily see that g cannot be extended to a unit in $A[z;\sigma]$, see Corollary 2.9. It remains an open question whether there are further restrictions on the degrees of the components.

5 Open Problems

We wish to close the paper with some problems open to future research. As described at the end of the last section, in the general situation it remains open as to which Forney indices (and complexity) a σ-cyclic code can attain. But from a coding theoretic point of view an investigation of σ-cyclic codes with respect to their distance is much more important. More precisely, it needs to be investigated whether one can relate the distance of a cyclic convolutional code to some properties of the generator polynomial (or any other suitable generating polynomial of the associated left ideal). As a starting point one might begin with minimal codes. In particular we think it is worth to investigate the construction of minimal codes via units as described in Corollary 3.7(2). Furthermore, it is also unclear which automorphisms should be chosen for obtaining good codes. Finally, the class of all cyclic codes of a given length needs to be investigated with respect to strong equivalence in the sense given in Remark 3.11. First ideas can be found in [9], they indicate that one may restrict to certain automorphisms in order to cover all equivalence classes. A detailed positive result would considerably reduce the amount of data to be investigated for the search of good cyclic codes.

References

[1] G. D. Forney Jr. Convolutional codes I: Algebraic structure. *IEEE Trans. Inform. Theory*, 16:720–738, 1970. (see also corrections in *IEEE Trans. Inf. Theory*, vol. 17,1971, p. 360).

[2] G. D. Forney Jr. Minimal bases of rational vector spaces, with applications to multivariable linear systems. *SIAM J. on Contr.*, 13:493–520, 1975.

[3] H. Gluesing-Luerssen and W. Schmale. On cyclic convolutional codes. Preprint 2002. Submitted. Available at http://front.math.ucdavis.edu/ with ID-number RA/0211040.

[4] H. Gluesing-Luerssen and W. Schmale. Distance bounds for convolutional codes and some optimal codes. Preprint 2003. Submitted. Available at http://front.math.ucdavis.edu/ with ID-number RA/0305135.

[5] T. W. Hungerford. *Algebra.* Springer, New York, 1974.

[6] R. Johannesson and K. S. Zigangirov. *Fundamentals of Convolutional Coding.* IEEE Press, New York, 1999.

[7] J. Justesen. New convolutional code constructions and a class of asymptotically good time-varying codes. *IEEE Trans. Inform. Theory*, IT-19:220–225, 1973.

[8] J. Justesen. Algebraic construction of rate $1/\nu$ convolutional codes. *IEEE Trans. Inform. Theory*, IT-21:577–580, 1975.

[9] B. Langfeld. Minimal cyclic convolutional codes. Diploma Thesis at the University of Oldenburg (Germany). Available at http://www-m9.ma.tum.de/dm/homepages/langfeld/thesis.pdf, 2003.

[10] F. J. MacWilliams and N. J. A. Sloane. *The Theory of Error-Correcting Codes.* North-Holland, 1977.

[11] J. L. Massey, D. J. Costello, and J. Justesen. Polynomial weights and code constructions. *IEEE Trans. Inform. Theory*, IT-19:101–110, 1973.

[12] R. J. McEliece. The algebraic theory of convolutional codes. In V. Pless and W. Huffman, editors, *Handbook of Coding Theory, Vol. 1*, pages 1065–1138. Elsevier, Amsterdam, 1998.

[13] P. Piret. On a class of alternating cyclic convolutional codes. *IEEE Trans. Inform. Theory*, 12:64–69, 1975.

[14] P. Piret. Structure and constructions of cyclic convolutional codes. *IEEE Trans. Inform. Theory*, 22:147–155, 1976.

[15] C. Roos. On the structure of convolutional and cyclic convolutional codes. *IEEE Trans. Inform. Theory*, 25:676–683, 1979.

[16] J. Rosenthal. Connections between linear systems and convolutional codes. In B. Marcus and J. Rosenthal, editors, *Codes, Systems, and Graphical Models*, pages 39–66. Springer, Berlin, 2001.

[17] J. Rosenthal, J. M. Schumacher, and E. V. York. On behaviors and convolutional codes. *IEEE Trans. Inform. Theory*, 42:1881–1891, 1996.

[18] J. Rosenthal and R. Smarandache. Maximum distance separable convolutional codes. *Appl. Algebra Engrg. Comm. Comput.*, 10:15–32, 1999.

[19] R. Smarandache, H. Gluesing-Luerssen, and J. Rosenthal. Constructions of MDS-convolutional codes. *IEEE Trans. Inform. Theory*, 47(5):2045–2049, 2001.

[20] M. Ventou. Automorphisms and isometries of some modular algebras. In *Algebraic algorithms and error-correcting codes; Proc. 3rd International Conf. AAECC-3*, pages 202–210. Springer Lecture Notes in Computer Science 229, 1985.